

CHILDREN WILL PLAY
by Phil Seifert

I am back this month despite playing what I considered a marathon game of MINER 2049er. Right now my high score for this game is 230,040. I fried out in Station 10 Zone 4. Boy, those little buggers get faster. If anyone out there has a higher score or comes within 20,000 points of mine, I would like to know about it so I could increase my margin.

Getting back to more important things, like giving you some game reviews. I have heard they have found a bug in the Atari Zaxxon. They have not started shipping as of this date, but, maybe they will have it out by the next meeting.

I am getting to sound like a broken record, but, WHERE IS MY STARBOWL FOOTBALL?! Gamestar was to start shipping these during the first week of March but do our loyal computer stores have it? Noooooooo.... We did see a demo disk of the game at our last meeting and it looks promising. I will have to see if the computer plays like the Washington Redskins or if it plays like the Minnesota Gophers football team.

One game I have got a hold of is Jumpman from EPYX. If you have a craving for climb the ladder games then this should fill that hole very nicely. The graphics are not as nice as MINER 2049er but the sounds are better and the game playability is there. If they had done a better job on the graphics I think this one would be a super game. But, don't let me stop you from buying this one, it is okay.

Another of those climbing ladders game has been released by Gebelli. It is called Candy Factory. I do believe those boys down there are getting better in their games for the Atari but this program doesn't measure up to Jumpman. If I had to choose, then go Jumpman. I have also seen Gebelli's Firebird and Embargo. These are both ROM cartridges and they remind me vaguely of Atari VCS graphics. My advice is to try these programs out in the stores, but, do not buy them sight unseen.

At our last meeting, we had a demonstration of a program called Hellcat Ace. Judging by the oohs and aahs this one might be popular. The company that makes this program is Microprose. If you weren't at the meeting, Hellcat Ace is a dogfight simulation with some fast graphics. I wasn't overly impressed but a lot of other people seemed to like it. Microprose also has two other games out; Chopper Rescue and Floyd of the Jungle. I have been told that Floyd is very interesting, but, I will have to see for myself.

Fort Apocalypse is now out from Synapse Software. This one fills my Choplifter empty spots. Here, you can rescue people and get points for shooting other things. This program, to me, is a Choplifter and Caverns of Mars all rolled up into one. I liked this game.

Sierra On-Line has released their Wall War program. What is this? This program has nice visual effects, but, the game play leaves something to be desired in my opinion. Try this one out at your favorite store and see what you think. I am used to a little better from Sierra On-Line.

Everybody remember ANALOG Magazine, the one that comes out when it feels like it? I am glad they felt like putting out two programs called Buried Bucks and Star Sentry. Buried Bucks has you in a helicopter with the mission of making money. Thar's gold buried in dem hills. From your chopper you drop bombs to clear away the dirt so you can go get it. Be careful, touching anything will blow you apart. There is one little obstacle, this airplane goes flying across the screen and dumps loads of new dirt, usually right where you are blasting a hole. I almost got addicted to this one last night but I managed to abstain. Check this one out. Star Sentry is another one of those in the Defender genre. It is okay, but I would stick with my Defender Cartridge.

Speaking of Defender, I am speaking to the son of Ed Aubitz, whom shall remain nameless, I hope your fingers fall off! In case the rest of you are wondering what I am talking about, here it is. This person has gotten a high score over 3,500,000 points. Can anyone beat that?

I have bought the first expansion to Hellfire Warrior called 'The Keys of Acheron' from EPYX. This gives my H.W. some added scenarios. They have released (finally!) Morlocs Tower. This is another adventure in the life of Brian Hammerhand. The programs are very nice, but I wish they would make enhanced versions that tell you what the treasure is instead of giving you numbers and give the room descriptions. I am getting tired of paging through their manuals for these. The manuals though, are very nice. I think they really get you into a mood for their adventure programs. The second expansion to H.W. called Danger in Drindisti should be out soon.

I finally got a chance to look at two cartridges from CBS-Kbyte Software called Krazy Kritters and K-Star Patrol. Of the two, I would rather have the kritters. But neither of these cartridges match the game playability of their first, Krazy Shoot-out. Oh well.

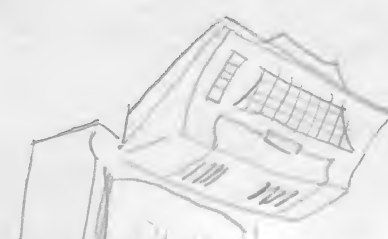
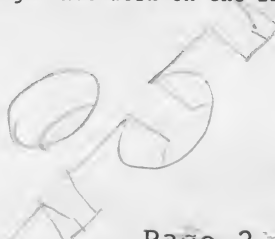
I thought you might like to know how I have been doing with my Chess program from Odesta. Well, I guess I am not that good a chess player. My middle game has a lot to be desired. If you like getting beat in Chess, Checkers, and Odin (othello) then I suggest you run out and buy these programs. In the near future, Jim and I are going to arrange a match between Odin and Reversal from Hayden. He maintains that Odin can't be tougher than Reversal. We'll see about that.

I am looking forward to buying my copy of Ultima I and II from Sierra On-Line if they ever manage to get them out. Look what happened to their Time Zone for the Atari. It disappeared into a time warp and was never heard from again. I have to admit, I am a little hooked on their Hi-res adventure games. It is really too bad I don't have the time to play them anymore. If you like adventures then I would recommend that you start with Wizard and the Princess. That should keep you busy for a while, unless you cheat.

While I am on the subject of adventures, I would like to point out that there is someone in our club who has completed all 12 of the Scott Adams adventures. That means I finally found someone who could help me in Savage Island part I & II. I am interested in collecting maps of various adventures and distributing them to club members. Right now I have; Shamus, Adventureland (S.A. #1), Pirate Adventure (S.A. #2), Kidnap from Softside, The Count (S.A. #5), Softporn Adventure, and Sands of Egypt. I should have fairly soon Wizard and the Princess and the other Scott Adams adventures. If anyone is interested in obtaining these maps, then call me (612-448-7042) or tell me at one of our meetings.

Everybody likes Shamus right? For those of you who can get through that thing at the highest level, I have some news for you. Synapse is coming out with a program called Shamus: Case II. I do not have the faintest idea what this one is about. Maybe it has some fancier stuff in it. Keep your eyes peeled for this one.

The last thing I want to talk about is the new Atari 1200XL. Atari said that if you follow their guidelines for programming then your software will run on the 1200. This is not turning out to be the case. Either an awful lot of software companies are using unauthorized jumps into the OS or Atari goofed. The percentage of software that doesn't work on the 1200 is pretty high. By the next meeting, I should have some information on this happening. In the meantime, if you have a 400 or 800 and are thinking of trading up to a 1200, wait, some of your software might not work on the 1200.



BITS and NIBBLES
by RAR

This is the first of what will hopefully become a regular column. The intent of this column is to review and report on the newsletters published by other Atari interest groups. I myself, as a relatively new Atari owner, am looking forward to this chance to review what others are doing.

One thing I've noticed in reviewing newsletters is that there are many very good writers out there as there are here in our group. So let's keep those articles and reviews coming in. I have always found that the best way to learn a subject is to write on it. Writing makes you really think about what you're doing.

Another interesting thing I've noticed is that some other groups are putting out newsletter special issues where all the articles pertain to one theme, such as printers, assembly language, etc.. If we can get enough interest on one subject and some willing people to write articles I think we should consider this for our newsletter. For those of you who have printers it may well be worth reviewing the February newsletter from The Jersey Atari Computer Group (JACG).

Just about all of the newsletters reviewed the new Atari 1200XLS. Many people were impressed with the packaging, but were a little wary since the 1200 seemed to be just a glorified 800 with a higher price tag.

The JACG newsletter mentioned that one of the original developers of the Atari 800, whose agreement not to come out with any peripherals for the 800 expires in October 1983, will come out with a super new peripheral during the last quarter of '83. It should be very interesting to see what happens.

The Little Rock Atari Addicts (LRAA) are having a very interesting contest going on. A One-Liner contest, where the best one line BASIC program wins the contest. The one line must be a meaningful, one logical line that is a totally self contained BASIC program. How about it? Should we try something like this?

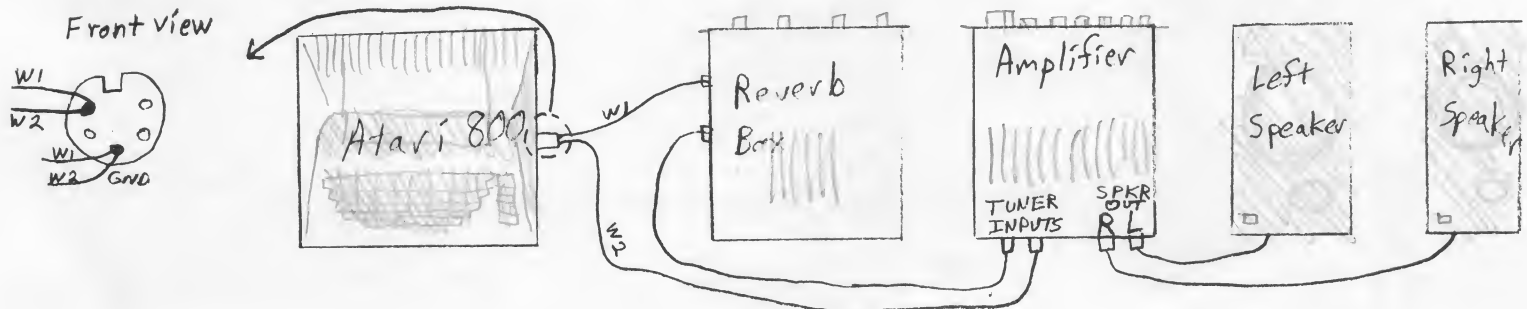
Have you seen Q*Bert at the arcades yet? Check it out. Apparently Parker Brothers has already bought up the rights to produce it for the VCS.

Well that's all for now.

SOUND
BY TRB

As a lot of you saw at the last meeting, the ATARI can produce some pretty impressive sounds when channelled through a stereo amplifier and a reverb unit. For those of you that missed the meeting, just imagine what STAR RAIDERS sounds like on your TV speakers, then imagine what the sound would be like given the full range (20 hz bass to 20000 hz treble) of a stereo system, and finally imagine what the sound would be like if you could add depth (deep cavern effect), repeats (echoing phasor shots), and delay (stereo effect) to one channel of the stereos' sound. If you still can't imagine what the sound is like, go down to ON-LINE computers and look at their set up (they have small speakers, but you will get the idea). I would like to thank Mike Davis (a TAIG member) for recommending this set up to me (he is heavily into music and is always trying things out).

Now for the fun part...what does it cost to configure your ATARI for super sound? Very little if you have a stereo system. I decided not to take over our family stereo (my wife would kill me, since I already hog her TV), so I went out and purchased an inexpensive amplifier (Realistic SA-102: \$60.00), a pair of speakers (Realistic 8"/2": \$40.00 each), 25 feet of 2 conductor PVC insulated telephone cable (\$2.50), one 5-pin Audio/Video Plug (#274-003: \$1.49), two RCA type PHONO PLUGS (#274-339: \$1.39 ea), some speaker cables to connect up the components (just used RCA plugs soldered to speaker wire), and finally the heart of the system, the reverb unit (Realistic Electronic Reverb: \$40.00). All of these items may be purchased from Radio Shack. The total cost of everything but the amplifier and speakers was around \$55.00. The following diagram indicates how everything should be hooked up.



SOFTWARE/HARDWARE NOTES
BY TRB

Well, not really too much new stuff this month (other than what Phil mentioned). QIX is out at last. I was disappointed in the speed at first, but it proves to be quite playable and addictive like the arcade version. I also got around to looking at File-Fax. This is the best data base program I have seen yet on the ATARI. It is very user friendly and easy to use (but best of all the databases can be expanded to multidrive configurations.) I was disappointed to see the incompatibility problems with the 1200 in so much software. I had to dump part of the 1200's OS to fix DISKEDIT and ULTRACOPY so that they would work on the 1200 (I deserved having to do so, because I did cheat a little in stealing vectors from the OS when I wrote the software.) There are some other problems with page 0 storage that I think will cause some bugs in other people's software, though.

Those of you who missed finding out what was on last month's disk, here is the listing.

Finance	financial aid program
Pong	machine language pong game
Frog	eat the flies before they get you
Bankshot	make some bankshots
Tinytext	a small word processor
Dungeon2	a small adventure program
Yahtzee	a game yahtzee anyone?

There is a small problem with the finance program. 1) I missed changing a 'RUN"D:INSTRUCT.BAS"' to a 'RUN"D:FINANCE.INS"'. I got one of them and didn't realize there might be another one. I suggest you change this. 2) The Finance program tries to run a program called 'PROPCOMP'. This file does not exist on the disk and I checked with the group I got the program from and they do not have the file either. This file involves property comparing. DO NOT USE THIS FEATURE.

Disk #17 (March 83) will have these programs.

Clock.dig	A digital clock
World	test your knowledge of world capitals
Dungeon.bat	another adventure
Football	try coaching for once
Speedread	sharpen your reading skills
Etchprint	A nice drawing utility with two pictures

I will put all but the football on the cassette. The file is 199 sectors long and there is more of a chance that it wouldn't copy right. I will attempt to convert the Etchprint to cassette but you will not get the extra picture files with it.

For new members and old ones that forget, you can get programs in these ways. 1) Buy a disk or cassette full of programs at the meetings or directly from Phil Seifert, 48 Kelley Rd., Chaska, MN 55318. 2) Order programs individually from Wayne Vasel (427-7922) or Jim Schulz (537-5442). 3) Make your own copies at the meetings.

Pre-made disks cost \$4.00 and cassettes \$3.00. All others no cost. If mailed to you, please include \$1.00 for mailing.

I am starting to have difficulty in finding new programs for our disks. I estimate I have enough for about 2 years. I would like all you adventurous programmers out there who will work for cheap (Free!) to write programs. We have to think about the future. The library will accept anything you personally wrote, not programs from magazines and books. I know it seems like it will be long time before I really need anything, but it would be nice to put things on our disk that our members wrote.

If you have questions ask me at the meetings or call Wayne or Jim. Thank you.

ADVENTURESOME PROGRAM by Mike Porter

I was lost in the desert and getting no where playing The Sands of Egypt. The Sands of Egypt is a great graphics adventure game with a little bit of animation. It is also very frustrating at times. I won the game as the door prize at the January TAIG meeting and I wanted to solve it by the next meeting. After I wandered in the desert and died of thirst a number of times I began to despair and began to look for ways to cheat. I could call Phil but I didn't want to admit that I was stuck so early in the game. I had managed to get to the cliff and the oasis a few times but it was pure luck and I didn't know how I had done it. There was water at the oasis but I was not allowed to drink without a canteen! The main problem is that you are wandering through the desert and there seem to be hundreds of places to be but they all seem to be about the same. I couldn't find anything to drop to see if I had been somewhere before and I was probably going around in circles. In my desperation I considered trying all possible combinations of the four directions north, south, east, and west for a set number of moves like 2 or 4 or something. Then I thought of writing a program which would compute all those combinations of directions and print them out so I could try them all without doing a combination over and over again. I could input the number of moves I wanted and the program would print them out on the printer. This could get pretty extensive (5 moves would give 1024 combinations)! Sound crazy? Well yes, but when you are dying of thirst in the desert you will grasp at any mirage. I finally did solve the adventure and escaped from the sands and I highly recommend the game and I didn't have to use trial and error. But I decided to try writing the program to give me the combinations anyway and so I thought I'd share it with any interested. I have also written a more general program where you input the BASE (as in BASE 2 or BASE 16, etc) and any decimal number and it will convert the number to the specified BASE but I'll leave that for another time. Now you see I figured that with 4 possible directions (N,S,E,W) it would be like counting in BASE 4. Each place or digit of a BASE 4 number has only 4 possibilities (0,1,2,3) so I would count in BASE 4 and convert 0 to north, 1 to south, 2 to east, and 3 to west. The program is below. In line 30 you input the number of moves. Then a string is dimensioned 1 greater than the number of moves and all characters of the string become N's (for North) except a blank as the last character. From line 80 on we count in decimal from 1 to the highest number possible in BASE 4 with N places (the number of moves input) and convert each number into BASE 4 then into the directions and print the combination to the screen. You could include Up and Down by changing BASE to 6 and adding the appropriate lines after line 180.

```

20 REM N=NUMBER OF MOVES
30 BASE=4 40 ? CHR$(125):? "NUMBER OF MOVES: ":INPUT N 50 DIM D$(N+1),A$(10): D$(1)="N": D$(N+1)="N":
D$(2)=D$ 60 D$(N+1,N+1)=" " 70 ? D$: 80 FOR I=1 TO INT(BASE^N*0.5)-1:Q=I 90 D$(1)="N": D$(N+1)="N": D$(2)=D$:
D$(N+1,N+1)=" "
100 FOR J=N TO 0 STEP -1
110 X=INT(Q/(BASE^J)):IF X>0 THEN D$(N-J,N-J)=STR$(X): Q=Q-INT((BASE^J)*X+0.5)
120 NEXT J
130 D$(N+1,N+1)=" "
140 FOR B=1 TO N
150 IF D$(B,B)="" THEN D$(B,B)="N"
160 IF D$(B,B)="1" THEN D$(B,B)="S"
170 IF D$(B,B)="2" THEN D$(B,B)="E"
180 IF D$(B,B)="3" THEN D$(B,B)="W"
190 NEXT B
250 ? D$:
300 NEXT I
350 ? : ? "RUN AGAIN?":INPUT A$:RUN

```


The following two pages include some useful information on updates to the ATARI BASIC manual and also some programming hints on writing software that will work on both the 800 and the 1200. Hopefully, some full sized sets of these pages will be available at the next meeting.
-ed



PRODUCT UPDATE

ATARI HOME COMPUTER SYSTEM

ATARI® BASIC Reference Manual Update

This product update contains a number of corrections and additions to the ATARI® BASIC Reference Manual.

Page 1. This definition is missing from the TERMINOLOGY list:

Page 6. This information pertains to the ARITHMETIC OPERATORS subtraction and exponentiation.

Page 7. This Note regards the use of the LOGICAL OPERATORS.

Page 13. This Note is in reference to SCREEN EDITING.

Page 20. This Note regards ON/GOSUB statements.

Page 22. Further information on RESTORE (RES.).

Page 25. Some additional information on using the INPUT (I.) statement:

Floating Point Number: A number containing an integer part, a decimal point, and a fractional part. The total number of significant digits in a floating point number, excluding the exponent, may be either nine or ten. This depends on whether the exponent is an even or odd multiple of 10.

Note: Avoid negating zero, as this will produce an invalid number. For example, if you type

```
PRINT -0
the result will be
-.0E+8
```

Note: Since the algorithm used to generate exponents, (x) is only an approximation, you cannot obtain integer results with it—for example, $2 \times 2 = 3.99999996$. To correct this, use the following technique:

```
X=2*2
PRINT INT (X+.5)
4
```

Note: Avoid using the statement PRINT A=NOT B, as the results are not predictable. Essentially, only PRINT statement with a NOT operator will be unpredictable.

Note: Large amounts of editing may lock up the system. It's recommended that programs under development be stored to cassette or diskette periodically (every 30 or 40 edits) with the SAVE or CSAVE command.

Note: If an ON/GOSUB expression evaluates to a number greater than the number of subroutine entries, then a POP statement will be necessary to clear the stack (see POP command, Section 4).

The RESTORE statement will not generate an error if the line number referenced does not exist. Instead it will RESTORE to the next larger line number in the program. Care should be taken to update RESTORE statements when renumbering a BASIC program.

When executing an INPUT from the screen, avoid moving the cursor away from and then back to the same line, otherwise, the wrong data may be input. Specifically, the INPUT prompt will be included in the INPUT string.

If a string of 128-255 characters is INPUT, then RAM locations 1536-1664 will be overwritten. This area is normally reserved for storage of programs or data. (See the ATARI Tech Reference Notes.)

To INPUT strings of more than 127 characters, use the GET command and store the values into a string (see Section 5, OPEN/CLOSE and PUT/GET commands).

Note: The maximum number of characters that can be INPUT from the screen is 120. The maximum for other devices is 255.

Note: Make sure that every INPUT statement has a variable after it; otherwise, unpredictable results may occur.



Page 26. This regards the use of the LOAD (LO.) command.

This Note should follow the LPRINT (LP.) command description.

Page 27. This information pertains to the file spec definition:

Page 28. This is an addition to the POINT (P.) section:

In the last paragraph under POINT (PR. or ?), the first sentence should read:

The following sentence should conclude the final paragraph on POINT (PR. or ?):

This note should then conclude this section on POINT (PR. or ?):

This Note regards the PUT (PU.)/GET (GE.) section:

Page 30. Here is a corrected version of the table—note in particular the correction on cmdno 22.

Note: If a program is loaded that is too large for the available memory space, it may give unpredictable results without an error message.

Note: An LPRINT command with a semicolon at the end will cause the following LPRINT statement to print on the next 40-column tab. A 40-column printer will move to the next line in such a case. To use the semicolon effectively, use the OPEN statement for the printer, then write to the printer with a PRINT statement (see OPEN/CLOSE and PRINT commands, Section 5).

Note: Be sure to include the closing quotation marks on a file spec parameter, especially when putting multiple statements on one line. For example,

```
OPEN #1, 4, 0, "D:TEST".STOP
will work, but
OPEN #1, 4, 0, "D:TEST.STOP
will not function correctly.
```

Note: To update a file, you must open it with a 12 in aesp1.

A comma tabs every 10 spaces.

However, if the last character to be printed (as in a string with quotation marks) is a **U** or **U**, then the next PRINT will begin at the end of the current line.

Note: In rare circumstances data printed to a diskette may have part of the BASIC program embedded in it. If this occurs, retry the operation.

Note: In certain circumstances the GET function may modify other variables within the program. To avoid this, PRINT any number to the screen between each GET.

cmdno	OPERATION	EXAMPLE
3	OPEN	Same as BASIC OPEN
12	CLOSE	Same as BASIC CLOSE
13	STATUS REQUEST	Same as BASIC STATUS
17	DRAW LINE	Same as BASIC DRAWTO
18	FILE	See Section 9
32	RENAME	XIO 32.#1,0,0,"D:TEMP CAROL"
33	DELETE	XIO 33.#1,0,0,"D:TEMP BAS"
35	LOCK FILE	XIO 35.#1,0,0,"D:TEMP BAS"
36	UNLOCK FILE	XIO 36.#1,0,0,"D:TEMP BAS"
37	POINT	XIO 37.#1,A,B
38	NOTE	XIO 38.#1,A,Y
254	FORMAT	XIO 254.#1,0,0,"D2"



Page 33. The last sentence in the paragraph about the CLOG function should read:

Page 34. The last sentence in the paragraph about the LOG function should read:

Page 38. The last line in the first paragraph should read:

Page 39. The first sentence should read:

In the second paragraph, the last line should read:

This is additional information on the VAL function:

This information pertains to String Concatenation.

In Figure 7-6, the correct result of the program on the left is:

Page 42. Some additional information on using the DIM (DI.) statement:

Page 43. This is an additional Note for the DIM (DI.) section:

Additional information on using the CLR command:

CLOG(0) through CLOG(1) are inaccurate and should not be used.

LOG(0) through LOG(1) are inaccurate and should not be used.

was stored there previously.

Upon execution, the screen displays THE SQUARE ROOT OF 10000 IS 100

number 1000000000.

Only the numeric field will be translated, while the text will be ignored. For example:

```
VAL("SSUM")=5
```

Note: BASIC cannot move strings of 256-character multiples correctly. String lengths should be checked; if any string contains a multiple of 256 characters, add or subtract one character from the amount to be moved.

BCD#

Make sure that the DIM statement does not contain a space between the string or array name and the left parenthesis of the dimensioned amount; otherwise, the following will happen—

```
DIM L (10) becomes DIM L10
```

—and this variable can no longer be referenced.

Note: The command COM is identical to DIM and may be used in its place.

Note: Due to a discrepancy in boundary checking, arrays of up to 32766 by 32766 in size can be dimensioned. The programmer should size the array ahead of time to ensure that there is no "virtual" storage space.

The second sentence in the last paragraph, beginning "It also clears ...," should be deleted.

The CLR command will not initialize the values in strings and arrays.



Page 45. Here is a corrected version of TABLE 9.1:

Page 49. The last sentence under PLOT (PL.) should read:

Page 50.

Page 51. The sentence directly under TABLE 9.4 should read:

Page 53. Here is a corrected version of TABLE 9.5.

TABLE 9.1—TABLE OF MODES AND SCREEN FORMATS

Gr. Mode	Mode Type	Horiz. (Columns)	Vert. (Lines) Split Screen	Vert. (Lines) Full Screen	Number of Color Registers	Split Screen	RAM Required (Bytes) Full Screen
0	TEXT	40	—	24	1h	—	952
1	TEXT	20	20	24	5	674	672
2	TEXT	20	10	12	5	424	420
3	GRAPHICS	40	20	24	4	434	432
4	GRAPHICS	80	40	48	2	694	690
5	GRAPHICS	80	40	48	4	1174	1170
6	GRAPHICS	160	80	96	2	2174	2164
7	GRAPHICS	160	80	96	4	4190	4200
8	GRAPHICS	320	160	192	1h	8112	8136

"The range of points begins at 0 and extends..."

In TABLE 9.3, the color PURPLE should be inserted after PINK in the first column, and the number 5 should be inserted after 4 in the second column.

"DEFAULT" occurs if no SETCOLOR statement is used.

MODE, SETCOLOR, COLOR TABLE

Default Colors	Mode or Condition	SETCOLOR (seap) Color Register No.	Color (seap)	DESCRIPTION AND COMMENTS
LIGHT BLUE	Mode 0 and all text versions	0	—	Character: luminance (same color as background)
DARK BLUE	—	2	—	Character: to be printed
BLACK	—	3	—	Border
ORANGE	—	4	—	Background border
LIGHT GREEN	Mode 1	0	—	Character: luminance
DARK BLUE	Mode 1 and 2	1	—	Character: to be printed
RED	—	2	—	Character: to be printed
BLACK	text modes	3	—	Background border
ORANGE	—	4	—	Background border
LIGHT GREEN	Mode 3, 5, and 7	0	—	Graphics point
DARK BLUE	—	1	—	Graphics point
BLACK	four-color modes	2	—	Graphics point (background default) border
ORANGE	—	3	—	Graphics point (background default) border
LIGHT BLUE	Mode 4 and 6	0	—	Graphics point
DARK BLUE	two-color modes	1	—	Graphics point
BLACK	—	2	—	Graphics point (background default) border
ORANGE	—	3	—	Graphics point (background default) border
LIGHT BLUE	—	4	—	Graphics point (background default) border
DARK BLUE	Mode 8	0	—	Graphics point (background default) border
BLACK	2 luminances	1	—	Graphics point (background default) border
ORANGE	—	2	—	Graphics point (background default) border
LIGHT BLUE	—	3	—	Graphics point (background default) border
DARK BLUE	—	4	—	Graphics point (background default) border
BLACK	—	5	—	Border



Page 54. In Figure 9-4, line 80 should read:

Page 55. This information pertains to TABLE 9.6:

Page 56. Here is a corrected version of TABLE 9.7

Page 58. The last paragraph should read as follows:

In TABLE 10.1:

Page 63. The last line in item 9 should read:

Page 67. In Figure 11-2, line 0260 under Data should be:

Page E-1:

Page H-7. Line 160 in the program should read

Page H-8. Line 50 in the program should read

Page 117.

Page 118.

Page 119.

80 X10 18, 06, 12, 0, "S."

In Column 1, # 14, a period, not a bar, shows on the screen

In Column 3, #'s 92-95 should show a superscripted circled 1 next to their characters.

TABLE 9.7—CHARACTER/COLOR ASSIGNMENT

	Column 1 Conversion	Column 2 Conversion	Column 3 Conversion	Column 4 Conversion
MODE 0	SETCOLOR 2 # = 32	# = 32	# = 4	NONE
MODE 1	SETCOLOR 0	# = 32	# = 32	# = 32
OR	SETCOLOR 1	NONE	# = 64	NONE
MODE 2	SETCOLOR 2	# = 160	# = 160	# = 96
	SETCOLOR 3	# = 128	# = 192	# = 128

2 Luminance controlled by SETCOLOR 1, 0, LUM

Note that the DATA statement in line 90 ends with 256, which is outside of the designated range. The 256 is....

The PITCH VALUE of 193 should have a musical note of "E," not "D."

precedence will save a few bytes.

#2

The right parentheses are missing after the word "CONSTANT" in Atan Functions of Inverse Cosine, Inverse Secant, and Inverse Cosecant.

160 IF K=125 OR K=155 THEN 180

50 PILOT 0,0,DRAWTO 159, DR

Following COM, "(see DIM)" should be deleted and replaced with "A-1."

Under "Input/Output Devices," Line Printer should be followed by "(P)," not "[L]."

"NOTE, 26" is missing from the listing



Page 120.

INSIDE BACK COVER here is the corrected table:

"STATUS, 29" is missing from the sublisting under "Statement" and also from the regular listing

MODE, SETCOLOR, COLOR TABLE

Default Colors	Mode or Character	SETCOLOR (see p. 1) Color Register No	Color (see p. 1)	DESCRIPTION AND COMMENTS
LIGHT BLUE	Mode 0 and all text windows	0	1	Color data actually determines character to be printed
DARK BLUE		2	3	Character luminance (same color as background)
BLACK		3	4	Background
ORANGE		4	0	Border
LIGHT GREEN	Mode 1	0	1	Character
DARK BLUE	and 2	1	2	Character
RED	(text modes)	2	3	Character
BLACK		3	4	Background
ORANGE		4	0	Background border
LIGHT GREEN	Mode 3, 5, and 7 (four-color modes)	0	1	Graphics point
DARK BLUE		1	2	Graphics point
RED		2	3	Graphics point
BLACK		3	4	Graphics point (background default) border
ORANGE	Mode 4 and 6 (two-color modes)	0	1	Graphics point
BLACK		4	0	Graphics point (background default) border
LIGHT BLUE		1	1	Graphics point luminance (same color as background)
DARK BLUE	Mode 8 (1 color, 2 luminances)	2	0	Graphics point (background default)
BLACK		4	—	Border

CONTINUED REV. A
© 1982 ATARI, INC.

APX Software Compatibility Guidelines

Violation of any of the following guidelines could cause a program to malfunction on the ATARI 1200XL. Programs written entirely in ATARI BASIC which do not use the PEEK or POKE instructions should have no compatibility problems. All other programs which do not deliberately attempt to violate one of the rules below are probably also compatible.

References are made in the following paragraphs to the Operating System User's Manual which is included with the ATARI Home Computer System Technical Reference Notes (CA016555, Rev. A, copyright 1982 ATARI, INC.). The notes can be ordered from your local ATARI Home Computer retailer or through ATARI by calling 800-538-8543 (in California 800-672-1404).

Note that the following paragraphs use the convention of preceding hexadecimal numbers with a "\$".

1) There are no "spare" locations in the operating system database. Memory locations \$0000-\$007F and \$0200-\$047F are reserved and should not be assigned to user variables or constants. For example, programs which assign values to the previously unspecified locations \$02C9-\$02E3 are likely to fail on the 1200XL.

2) "User accessible" variables (such as the left margin of text area, LMARGN at \$0052, and the cursor inhibit, CRSINH at \$02F0) have retained their original addresses. However, "internal" and "temporary" variables may have been moved, so no attempt should be made to access them. For example, PBUF\$2, previously at \$02DF, has been relocated.

Warning: NEWROW and NEWCOL, previously at \$0060 and \$0061, respectively, have been relocated. They are internal variables even though they are specifically referenced in the ATARI 400/800 BASIC Reference Manual.

3) System calls to operating system routines should use only "advertised" entry points and vectors. These addresses are provided in Appendix J of the Operating System User's Manual. Vectors labeled "for OS internal use only" (such as DISKIV), or found from scanning the operating system listing may have moved and should not be used.

4) No attempt to detect BASIC, PILOT, or other types of cartridges should be made. Since no documented procedures exist for doing so, revisions to these cartridges could cause such programs to fail. Similarly, no attempt should be made to execute a jump directly into cartridge ROM.

5) Programs should conform to the interfaces described in the Operating System User's Manual with regard to:

- Power up initialization (coldstart).
- RESET initialization (warmstart).
- Interrupt processing.
- System timer utilization.
- I/O operations.
- Floating point operations.
- Memory management.

6) Programs should not rely upon "side effects" of system functions. The interface for each operating system function is documented; since "bonus" features such as finding a useful constant in a register upon return cannot be relied upon, they should not be used.

7) The "initialization values" given in section 6 of the early versions of the Operating System User's Manual are not valid for the 1200. They should not be relied upon.

8) No attempt should be made to access the hardware register PORTB, at memory location \$D301 (\$4017 decimal). Since the two game ports formerly controlled by PORTB have been eliminated from the 1200, it has been reallocated for other purposes.

NOTE: Many popular computer magazines and textbooks provide sample programs which break some of the guidelines presented above. One common example is the provision of so-called "useful POKE locations" for ATARI BASIC programmers without any mention of which variables are internal. Do not assume that a program will run on an ATARI 1200XL if it uses a technique from a published source that breaks any of the preceding guidelines. Use the Operating System User's Manual as the authoritative source if there is any doubt about the legality of a specific programming decision.

DISK ARCHIVING by TRB

After fielding some questions recently about what is really out there on DOS and game disks, I realized that a lot of new ATARI owners don't have easy access to some of the magazine articles/tutorials that were so helpful in teaching me about the disk system. With this in mind, I am going back to the basics this month by including the following section from my DISKEDIT documentation.

....This is the first of hopefully many addendums that will address the fundamentals of the DISK system as well as some more advanced concepts. I have decided to devote the first section to disk storage, since this subject is glossed over in most documentation I have seen available to date. If you have further questions about the Disk system after reading this material, I strongly recommend that you obtain the ATARI technical user notes for reference.

DISK STRUCTURE:

A diskette that has been formatted on an ATARI 810 disk drive contains 40 tracks. Each track circles the disk in the form of a ring and is composed of 18 sectors. Each sector is composed of 128 bytes of information. A disk can contain, therefore, $40 \times 18 \times 128 = 92160$ bytes of information (a byte can be thought of as one character representing a value between 0 and 255-00 to FF hex). The diagram to the right depicts this structure in detail.

Note that this will be the last I talk about tracks. The disk structure is best thought of as having 720 sectors (40×18), since this is the way they are addressed by the ATARI computer. Because of a discrepancy in the ATARI OS, we will think of these sectors as being numbered 1 to 720, even though ATARI DOS can only read sectors 1 to 719.

SECTOR TYPES:

I have discussed before the types of bad sectors that exist, so I will now describe the types of 'good' sectors that you will encounter when looking at a disk. Aside from empty sectors (sectors containing all zeroes), there are five main sector types-autoboot (BOOT), DOS file (FILE), non DOS file (NOTF), volume table of contents (VTOC), and directory (DIR) sectors.

A boot sector is a sector that is automatically loaded when the computer is turned on. All DOS or game disks have one or more boot sectors in a contiguous block of sectors beginning with sector 1. In DOSII disks, sectors 1-3 are the boot sectors (these in turn load in the DOS.SYS binary program and execute it.) In non DOS (i.e. game) disks, all that may autoloading in is sector 1 (which in turn loads in the rest of the game), or all of the game may load in as part of the boot.

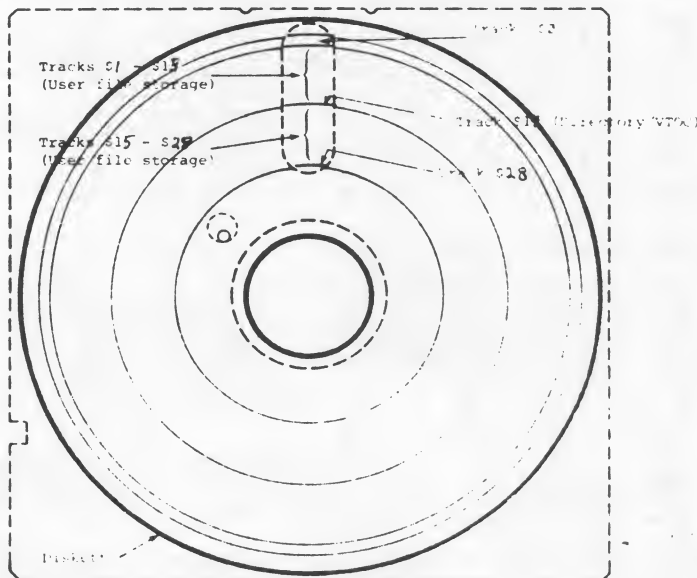
A NOTF sector is any sector on a disk that is accessed directly by sector number via the disk handler (via a call to the OS through SE453 for example). In other words, this type of sector only exists on a game boot disk or on a non standard DOS disk, since DOS can not be used to read or even create it.

The final 3 sector types (DOSF, DIR, and VTOC) are found on DOS disks. The DOSF file is the file that contains your programs and databases. DOS stores files on a disk in such a manner that it can always look at any sector of a file and determine which file the sector belongs to and what sector is next in the file chain. This capability is not without some cost however, since not only data must be stored on a DOSF sector, but also some bookkeeping overhead is required. A DOSF sector thus will contain only 125 bytes of user information followed by 3 bytes that will indicate the file index #, the next sector # in the chain, and the # of usable bytes in the sector. DOS keeps track of which files are stored where by using a set of sectors called the directory. Each DIR sector contains 8 index locations for file information (16 bytes/file). Each location contains information on the name of the file (bytes 6 to 16), the starting sector number of the file (bytes 4 and 5), the size of the file (bytes 2 and 3), and the status of the file (the first byte). Eight sectors are reserved on a DOS disk for the directory info (sectors 361 to 368). Finally, the VTOC (volume table of contents) sector is the sector that contains information regarding which sectors on a disk are used and how many are free.

All three of the DOS file types are used for any DOS file I/O operation. For example, let's say that you want to store a BASIC program to disk. When you do a SAVE of the program, DOS assigns the file a location in the first available location in the directory sectors. DOS then scans the VTOC directory to find the first available sector to start storing the data to. This sector number is placed in the file directory sector (bytes 2 and 3) and then 125 bytes of your file is transferred to this sector along with a trailing 3 bytes. DOS then updates the VTOC and continues the scan, transfer, and update process until all of the file is transferred. When done, the file status byte is updated in the directory sector to indicate that the file is good and closed.

FINDING FILES:

Now that you know something about how the disk is structured, the only thing left is to go through a quick example of how to find a file on a disk. The assumptions made in this example are that you have just SAVED a BASIC program to a newly formatted disk which has DOS and DUP written to it. If you look at sector 361, you will see that your file is the third entry in the directory table and that it starts at sector number 85 (byte 4 at the file location equals 85). If you look at sector 85 now, you will see the start of your program (in tokenized form of course.) In the following sample cases, I hope to give you a better understanding of the differences between standard DOS disks and most autoboot game disks.



SAMPLE #1: A newly formatted DOS II disk, with DOS&DUP.

SECTOR #	TYPE	DESCRIPTION/FUNCTION
1	BOOT	Tells computer to autoread sectors 1-3 into mem @ \$700
2	BOOT	Once this three sector boot program is put into memory,
3	BOOT	the computer begins executing it at \$706
4	FILE	DOS.SYS. This is loaded and executed by the boot program.
.	FILE	DOS provides the user the ability
42	FILE	to access (SAVE,LOAD,OPEN,etc) disk files.
43	FILE	DUP.SYS. This gives the user file deletion, copying,
.	FILE	and other file managment functions.
84	FILE	
85	FILE	START of available file space for user programs.
FILE		The first file stored on a disk after writing DOS&DUP
359	FILE	will be stored at the lowest sector number available.
360	VTOT	Tell DOS what sectors are available for storage.
361	DIR	These 8 sectors tell DOS where each file
.	DIR	is stored. Each sector contains 8 file entries
368	DIR	of 16 bytes each (giving you 64 file locs total).
.	FILE	More file storage space.
720	FILE	This sector is accessible only via the DISK handler.

SOFT UNLIMITED'S
SUPER SALE

Specials of the Month
(order before 5/1/83)

ITEM	SRP	SALE	%OFF
WAYOUT	40	26	35%
MAC65	80	55	31%
BASIC A+	80	55	31%
SPEED READ +	60	38	37%
BANDITS	35	24	31%
THRESHOLD	40	27	33%
VISI-CALC	250	175	30%
MINER 2049ER	50	38	25%

INCLUDE \$1.50/ITEM FOR SHIPPING
(and don't forget sales tax)

Send check or money order to:
SOFT UNLIMITED
3546 PILGRIM LN
PLYMOUTH, MN 55441

DISKEDIT OWNERS: CARTDISK + is now available for \$20.00. This is a fully machine language cartridge copier that will automatically search out most cartridge protection schemes. The preliminary version interfaces to DISKEDIT only, so this program will be sold only to DISKEDIT owners. 2.0 has been upgraded again, so stop by at the meeting for a free trade (\$5.00 for 1.0 owners).

SAMPLE #2:A boot game-non DOS.

SECTOR #	TYPE	DESCRIPTION/FUNCTION
1	BOOT	Tells the computer to boot the next 6 sectors.
2	BOOT	This is the continuation of the boot routine.
3	BOOT	Once loaded, this routine will begin executing
4	BOOT	and load sectors 257-400 into memory. Execution
5	BOOT	will then pass to the game code. Note that
6	BOOT	these 6 sectors may contain a bad sector check.
7-255	----	Empty sectors.
257	NOTF	This is the game code.
.	NOTF	Note that these sectors may also
400	NOTF	contain a bad sector check.
401-450	----	More empty sectors.
450-720	----	Bad (unformatted) sectors.

Assume that sector 1 used in case 2 contained the following first six bytes: byte 1=0 (normal noisy load), byte 2=6 (load 6 sectors), byte 3=0 and byte 4=6 (start loading into memory at address \$600), and bytes 5=6 & 6=6 (set the system reset vector to address \$606). This is just for example purposes only. It is best to just look at a game disk and play around with it.

computer FOOD

Software for Personal Computers

**We sell software at prices
that won't take a big byte out of your wallet.**

Here's just a sample from our
large menu of COMPUTER FOOD:

BANDITS.....\$23.97*	MIDNIGHT MAGIC..... 25.22
BASIC COMPILER..... 67.47	SEAFOX 21.97
CANYON CLIMBER..... 21.97	TEXT WIZARD 67.47
CHOPLIFTER 25.22	WAYOUT 26.97*
GHOST HUNTER..... 21.97	
HODGE PODGE 15.47	ZAXXON 27.97*

For further information or prices call
(612) 560-2917

To order send check or money order to:

Steve Pauley
COMPUTER FOOD
2547 - 83rd Court No.
Brooklyn Park, MN 55444

Include \$1.50 per item for shipping plus 6% sales tax.

You can save shipping cost by picking up software at the address above (call first) or from me at the next TAIG meeting.

Most programs will be mailed or can be picked up within 1 to 4 days of receipt of your order.

*This price will be going up soon.

ETC.

You remember the Atari booster buttons? Hey, the fun we had with the E.T. dolls? I'll never forget the laughs I had with my Atari joy buzzer. And I never go anywhere without my Official Atari decoder ring. Now this:

ATARIAN HANDSHAKE

1. Clasp hands as in a regular handshake.
2. Lift the other person's thumb and grasp it with your free hand.
3. Rotate the thumb you're holding, making believe it's a joystick.
4. Enjoy.



Thanks to the Atari Boosters League East - Winter Park, Florida, for this gem.



12959 RIDGEDALE DRIVE, MTKA.
RIDGE SQUARE SOUTH
(WEST OF RIDGEDALE MALL)

546-8814
10-9 00 (M-F) 10-5 00 (SAT) 12-5 (SUNDAY)

SALE

ON LINE COMPUTER CENTERS OF MINNEAPOLIS
COMPUTER FAIR
FRIDAY, MARCH 25 - SUNDAY, MARCH 27

SALE

- I. 20% off all software in stock
- II. 20% off all joysticks, paddles and other accessories in stock
- III. 15% off computer furniture in stock or by special order
- IV. PRINTERS:
 - * EPSON FX-80 (1 box of paper with purchase)
 - * COMWRITER (in stock or special order only \$995.00)
 - * DAISYWRITER (in stock or special order only \$1395.00—includes 48K buffer.
- V. MONITORS:
 - * ALL GREEN AND COLOR MONITORS (Comrex, Panasonic, Taxan, and Zenith) 15% OFF.
- VI. COMPUTER SYSTEMS:
 - * ATARI 800 SYSTEM (inc. computer & disk drive) ONLY \$ 995.00.
 - * ATARI 1200 SYSTEM (inc. computer & disk drive) ONLY \$ 1299.00

ON LINE RENTAL CLUB

ON LINE COMPUTER CENTERS INTRODUCES THE SOFTWARE RENTAL CLUB AND INVITES ALL ATARI INTEREST GROUP MEMBERS TO TAKE ADVANTAGE OF THE LARGE SELECTION OF RENTAL SOFTWARE NOW AVAILABLE FOR THE ATARI 800 AND 400 COMPUTERS.

- * \$25 LIFETIME MEMBERSHIP FEE
- * REASONABLE RATES ON A PER DAY BASIS
- * DEPOSIT REQUIRED AT COST OF SOFTWARE BEING RENTED
- * 100% OF DEPOSIT MAY BE APPLIED TO THE PURCHASE OF THE SOFTWARE RENTED
- * OVER 150 PIECES OF SOFTWARE TO SELECT *
- * CONSTANT UPDATING AND SELECTION
- * MUST BE 18 YEARS OF AGE TO RENT

ATARI HOME COMPUTERS
We've Brought The Computer Age Home.™

(ATARI GROUP DISCOUNTS DO NOT APPLIED TO THE
RENTAL CLUB NOR TO ON LINE SALE PRICES)





PRAIRIE MICROSYSTEMS, INC.

Software Development / Computer Sales - Services

941-4504

MEMORY EXPANSION SPECIALS

NOW IS THE TIME TO EXPAND YOUR 400'S OR 800'S MEMORY WITH A BOARD FROM INTEC PERIPHERALS. ADDING TO THEIR COLLECTION OF 16, 32, AND 48K RAM BOARDS, INTEC HAS JUST ANNOUNCED A 64K BOARD FOR THE 400. LIKE THE HIGHER-PRICED BOARDS, THIS ONE USES 64K CHIPS AND BANK-SELECTION TO EXPAND THE 400 COOLY AND EFFICIENTLY ... AND JUST LOOK AT THE PRICE!

PRAIRIE PRICE FOR THE NEW 64K BOARD:

\$130!!

INTEC HAS ALSO LOWERED THE PRICE OF ITS 48K BOARD FOR THE 400. PRAIRIE'S NEW LOW PRICE:

\$ 97!!

THE PRICES ON THE 32 AND 16K BOARDS FOR THE 400 OR 800 REMAIN THE SAME, AT \$70 FOR THE 32K AND \$51 FOR THE 16K.

ALL INTEC PRODUCTS ARE EASILY INSTALLED AND COME WITH A LIFETIME WARRANTY.

* * * * *

SOFTWARE CLEARANCE

TO REDUCE OUR INVENTORY, PRAIRIE IS OFFERING THIS MONTH SOME GREAT DEALS ON SELECTED SOFTWARE, WHILE SUPPLY LASTS.

³⁵
** ~~30~~% OFF THE FOLLOWING ITEMS **

ASTERIODS (CART), MUSIC COMPOSER (CART), MISSILE COMMAND (CART), SPACE INVADERS (CART), MY FIRST ALPHABET (D), SALMON RUN (D), SCRAM (C), THE JAR GAME/CHAOS (D), GOLF CLASSIC/COMPUBAR (D), THE BIG MATH ATTACK (C & D), MICROPainter (D)

³⁰
** ~~28~~% OFF THE FOLLOWING ITEMS **

FANTASTIC VOYAGE (CART), SNEAKERS (D), CANYON CLIMBER (C & D), PREPPIE (D), CHOPLIFTER (D), PICNIC PARANOIA (D), CAVERNS OF MARS (D), PROTECTOR II (D), DAVID'S MIDNIGHT MAGIC (D), LABYRINTH (D), SEAFOX (D), BANDITS (D), STELLAR SHUTTLE (D), APPLE PANIC (D)

* P R A I R I E M I C R O S Y S T E M S * 9 4 1 - 4 5 0 4 *

PICK UP YOUR COPY OF OUR MARCH NEWSLETTER/CATALOG AT THE NEXT TAIG MEETING, OR CALL TO BE PLACED ON OUR MAILING LIST.

8711 Black Maple Dr. Eden Prairie, MN 55344

TRANSLATING OTHER BASICS INTO ATARI BASIC

by William Frank
Reprinted from H.A.C.E.
(Houston A.C.E.)

1. DEF(X)=INT(RND(0)*7.98+1.01)
ATARI has no DEF so you either have to define the function each time it appears or use a GOSUB.
2. IF...THEN..ELSE
ATARI has no ELSE so you need to make the ELSE part the next line.

3. IF...THEN
In ATARI Basic if the condition is not met, control passes to the next line. Use multiple statements on the same line # as an IF statement only if they are also conditional.

4. LEFT\$(A\$,Y) returns the Y leftmost characters of A\$
ATARI has no LEFT\$ so you need to use A\$(1,Y).

5. MID\$(A\$,X,Y) returns Y characters of A\$ starting with the X character.
ATARI has no MID\$ so you need to use A\$(X,X+Y-1).

6. RIGHT\$(A\$,Y) returns the Y rightmost characters of A\$.
ATARI has no RIGHT\$ so you need to use
A\$(LEN(A\$)-(Y-1)),LEN(A\$)).

7. A\$(I) means I strings.
ATARI has no string arrays. Use A\$(I-(B-1),I*B) where B is the length of the longest string.

8. A\$=A\$+B\$
A\$(LEN(A\$)+1)=B\$

9. CLS means clear the screen
GR.0

10. TEXT means enter Text mode
GR.0

11. HOME positions cursor on Apple
POSITION 2,0

12. PRINT TAB(12);"HELLO" means print HELLO 12 spaces over
ATARI has no TAB function so use
POSITION PEEK(84),PEEK(85+12)
PEEK(84) is current row
PEEK(85) is current column

13. VTAB 12
ATARI has no VTAB. Use POSITION PEEK(84)+12,PEEK(85)
14. HLIN A,B AT C Apple Horizontal Line.
PLOT A,C:DRANTO B,C
15. VLIN A,B AT C Apple Vertical Line.
PLOT C,A:DRANTO C,B
16. SET A,B (Apple)
PLOT A,B
17. RESET A,B (Apple)
COLOR 0:PLOT A,B
18. PRINT @ 234 TRS-80 Position Statement
Use POSITION STATEMENT.
[Editor's Note: Line=pos/40, Col=pos-(line*40)]
19. PRINT USING###.##;X TRS-80 format allowing only two decimal places.
X=(INT(X*100))/100;PRINT"\$";X
20. NUMBER# # means double precision (12 places instead of 6 on TRS-80)
No equivalent.
21. INPUT "How many turns";X
PRINT "How many turns";:INPUT X
22. INPUT A(3)
INPUT Z:A(3)=Z
23. RND(3) On some computers a random number from 0 to 3.
RND(0)*2 ATARI only gives zero to one.
24. ON ERROR GOTO 1000
TRAP 1000

As in all life nothing is totally free. The ATARI is far superior to other computers in having three microprocessors instead of one. This is what allows for the fantastic sound and graphics. To allow for the sounds and graphics and not have BASIC take up more than 8K, some minor sacrifices were made. The above shows that there are not too difficult ways around all of the omissions from the ATARI BASIC. I for one am convinced we have gained far more than we have lost!!!

[Editor's note: I am also convinced we gained more than we lost, especially when you consider the ability to abbreviate commands and the immediate syntax checking. Many of the non-ATARI BASIC commands in this article, are included in BASIC AT. We will have a review of BASIC AT next month.]

valFORTH for the ATARI 400/800

Professional Software for the Hobbyist

Forth has been used for years by ATARI and others in programming their arcade games. FORTH is fast, 15-20 times faster than BASIC, and can make use of every capability for your computer. And it's no longer the province of the professional programmer! With valFORTH and the additional packages available at WIZARD'S WORK, you can create programs in an afternoon that would have taken weeks of hard work!

valFORTH alone requires 24K
valFORTH plus one or more packages requires 32K minimum.
Memory Requirements Include 10-12k Working Space

Wizard's Work
9438 36th Ave. No.
New Hope, MN 55427
545-2136

MINER 2049er NUTS!!!!
meet at WIZARD'S WORK
THUR. NITE 3/31/83 OR
4/7/83 from 6 to 8 PM
(Learn to advance out
of the MINE SHAFT)
compare notes & scores
with the experts!!!!

**Wizard's
Work** 
your home computer center

TWINCITY ATARI INTEREST GROUP
6824 QUEEN AVE. S.
RICHFIELD, MN 55423



Next TAIG Meeting:
Sunday, March 27, 1983
TAIG - 7:00 P.M.

Odd Fellow's Lodge
3003 27th Ave. S.
Mpls, MN